Environmental Modelling and Software 112 (2019) 143–156

Contents lists available at ScienceDirect

Environmental Modelling & Software

journal homepage: www.elsevier.com/locate/envsoft

The MESSAGE$_{ix}$ Integrated Assessment Model and the *ix modeling platform* (ixmp): An open framework for integrated and cross-cutting analysis of energy, climate, the environment, and sustainable development☆

Daniel Huppmann*, Matthew Gidden, Oliver Fricko, Peter Kolp, Clara Orthofer, Michael Pimmer, Nikolay Kushin, Adriano Vinca, Alessio Mastrucci, Keywan Riahi, Volker Krey

Energy Program, International Institute of Applied Systems Analysis (IIASA) Schlossplatz 1, 2361, Laxenburg, Austria

ARTICLE INFO

Keywords:
Open-source
Energy system optimization
Integrated assessment
Strategic planning tool
Scenario management
Data warehouse*JEL classification:*
Q40
Q50
C61

The MESSAGE Integrated Assessment Model (IAM) developed by IIASA has been a central tool of energy-environment-economy systems analysis in the global scientific and policy arena. It played a major role in the Assessment Reports of the Intergovernmental Panel on Climate Change (IPCC); it provided marker scenarios of the Representative Concentration Pathways (RCPs) and the Shared Socio-Economic Pathways (SSPs); and it underpinned the analysis of the Global Energy Assessment (GEA). Alas, to provide relevant analysis for current and future challenges, numerical models of human and earth systems need to support higher spatial and temporal resolution, facilitate integration of data sources and methodologies across disciplines, and become open and transparent regarding the underlying data, methods, and the scientific workflow.

In this manuscript, we present the building blocks of a new framework for an integrated assessment modeling platform; the "ecosystem" comprises: i) an open-source GAMS implementation of the MESSAGE energy++ system model integrated with the MACRO economic model; ii) a Java/database back-end for version-controlled data management, iii) interfaces for the scientific programming languages Python & R for efficient input data and results processing workflows; and iv) a web-browser-based user interface for model/scenario management and intuitive "drag-and-drop" visualization of results.

The framework aims to facilitate the highest level of openness for scientific analysis, bridging the need for transparency with efficient data processing and powerful numerical solvers. The platform is geared towards easy integration of data sources and models across disciplines, spatial scales and temporal disaggregation levels. All tools apply best-practice in collaborative software development, and comprehensive documentation of all building blocks and scripts is generated directly from the GAMS equations and the Java/Python/R source code.

1. Introduction

Numerical tools for energy-economic-engineering-environment (E4) systems and "Integrated Assessment Models" (IAM) are a vital component in the analysis of energy system transitions in the context of climate change mitigation and sustainable socio-economic development. These tools are applied to advance scientific understanding of the underlying dynamics, and to evaluate the various policy options to mitigate climate change, safeguard the environment, and ensure universal access to clean and reliable energy (e.g., Riahi et al., 2017; Edenhofer et al., 2014; Riahi et al., 2012).

It is an often-repeated mantra of applied research that "modeling is for insights, not numbers" (cf. Huntington et al., 1982). Indeed, numerical optimization and simulation tools are well suited to illustrate the interdependencies within and between the complex human and earth systems, highlighting interactions and feedback effects that may seem counter-intuitive at first glance. In addition to qualitative insights, numerical quantifications of pathways or storylines serve a useful role in illustrating the underlying narrative. Even with the caveat that specific parameters are not known with the desired level of confidence, parametrizing a model requires the researcher to carefully and deliberately choose values that are a best estimate at the time of conducting

☆ The authors would like to thank Alexander Koeberle, Simon Parkinson, Victor Maus, David McCollum, Shinichiro Fujimoi, Mariana Imperio and Miguel Poblete-Cazenave for feedback and discussions during development of the framework.

* Corresponding author.
E-mail address: huppmann@iiasa.ac.at (D. Huppmann).

https://doi.org/10.1016/j.envsoft.2018.11.012
Received 12 March 2018; Received in revised form 7 November 2018; Accepted 26 November 2018
Available online 29 November 2018
1364-8152/

the analysis (Paltsev, 2017; Weyant, 2014).

1.1. Three paradigm shifts of systems modeling

Numerical models have long been a cornerstone of the dialogue between the scientific community and the global policy arena. Alas, there are numerous fundamental shifts with regard to the analysis of energy systems and the environment in a context of sustainable development and climate change. These new requirements are driving developments of numerical models for human and earth systems (e.g., Pfenninger et al., 2014).

Below, we highlight three of these shifts to illustrate how the framework introduced in this manuscript can facilitate a more effective analysis to ensure that numerical modeling can provide adequate answers to current and future challenges.

1.1.1. Increasing complexity

First, the systems that numerical tools are expected to describe are exhibiting ever *increasing complexity*. This can be illustrated in the electricity sector, where variable renewable energy sources are requiring a paradigm shift in the approach to energy investment, planning and market design (Pietzcker et al., 2017; Johnson et al., 2016; Ueckerdt et al., 2015). Furthermore, the quest for a better representation of complex interactions goes beyond the technical-engineering aspects and into the realm of social sciences. For example, Rao et al. (2014) explore the interdependence between emissions and poverty eradication, and Sohail et al. (2017) analyze synergies and trade-offs between energy-efficient urbanization and health. While a broad-brush approach centered on an assumption of rising incomes and associated energy demand growth may imply ever-increasing emissions, a more focused analysis centered on human development and decent living can identify opportunities to satisfy basic needs globally without exceeding available resources.

1.1.2. Nexus issues and interaction across sectors

Second, improved awareness of *nexus issues and interaction across sectors* draws attention to methodological limitations and overly-constraining model boundaries. Furthermore, considerations of cross-sectoral impacts may render options or solution strategies infeasible that seemed to be the preferred choice in a one-dimensional, narrowly focused analysis (cf. Keairns et al., 2016).

While the first challenge can be overcome by developing models that are more detailed with regard to spatial and temporal resolution, adequately addressing nexus issues usually requires linking models from various disciplines including engineering, geophysics and earth sciences, as well as economics and the social sciences. These scientific fields usually apply different methodologies, making the consistent communication of assumptions and an effective integration between them a non-trivial matter (Mantzos et al., 2016).

A further consequence of these considerations is a paradigm shift away from monolithic models towards a bifurcation of model development: on the one hand is a modular design of frameworks, which provides flexibility and facilitates updating both the input data and the mathematical formulation or structure underpinning individual modules of ever more complex assessment tools. The EIA's National Energy Modeling System (NEMS) is a case in point.[1]

The alternative to such groups of "soft-linked" in-depth sectoral models is the development of a deliberately stylized representation of drivers and impacts. By virtue of nimble and computationally efficient design, such models can be run much more often than complex modular frameworks. In this way, such stylized models can provide valuable first-order approximations of interdependencies and trade-offs. The reduced-complexity climate model MAGICC (Meinshausen et al., 2011) is an example of the latter approach. It is frequently coupled with integrated assessment models to evaluate temperature implications of energy systems transition and climate change mitigation scenarios.

1.1.3. Transparency, reproducibility, and intelligibility

The third driver for a new modeling paradigm is the need for *transparency, reproducibility, and intelligibility* of scientific analysis – both with regard to the methodology as well as the underlying data and assumptions. IAMs in particular have come under scrutiny for their "black-box" nature and the perceived ad-hoc approach to important aspects (Rosen and Guenther, 2015; Pindyck, 2013; DeCarolis et al., 2012). More importantly, it is a fundamental caveat of long-term, numerical energy-economy models that they cannot be validated against real-world results – at least not in the sense of validation applied in climate research or other natural sciences tools based on invariant physical principles (Wilson et al., 2017a).

In response to increasing focus on validation and critical assessment, policymakers, funding agencies and the scientific community have increased their expectations of openness, reproducibility and standardization of model-based policy analysis (Pfenninger et al., 2017; Cao et al., 2016). Furthermore, the development of open-source software and tools for effective collaboration has revolutionized many fields of business and academic analysis over the past decades. This new paradigm of openness has also reached the energy and climate change research community, initiating numerous new projects and initiatives (e.g., Morrison, 2018; Pfenninger et al., 2018).

1.1.4. The need for a new kind of modeling framework

To effectively tackle the issues surrounding transparency, reproducibility and intelligibility, a paradigm shift is required that goes beyond implementing marginal extensions in established modeling platforms like MESSAGE (see next section), TIMES (Loulou and Labriet, 2008), or OSeMOSYS (Howells et al., 2011). Instead, a new infrastructure is required to seamlessly integrate scientific workflows, data processing and policy analysis across different disciplines and spatial scales. This modeling system must draw on best-practice in collaborative software development and apply the best methods for different parts of the "scientific supply chain", ranging from data collection via pre-processing and numerical computation to a comprehensive toolbox for analysis and evaluation of the results.

1.2. The history of the MESSAGE model

The *Model for Energy Supply Systems And their General Environmental impact* (MESSAGE) is a *process-based integrated assessment model*; it allows for a detailed representation of the technical-engineering, socio-economic, and biophysical processes in energy and land-use systems. It is a linear/mixed integer optimization model, aiming to satisfy a given demand level at least cost, with optional coupling to a stylized macro-economic model to describe the feedback of end-use prices on demand for energy services.

The model has been developed at IIASA over the last four decades (Schrattenholzer, 1981). While the first version focused on the supply-side of fossil resources and nuclear energy, subsequent work extended the mathematical formulation and the scope of the full energy system representation (Messner and Strubegger, 1995). Pioneering work on endogenizing technological change into an energy system model was done with MESSAGE (Messner, 1997). Further, the least-cost energy system optimization model was linked to the general-economy model MACRO to incorporate feedback from end-use prices on energy demand and substitution between different sectors (Messner and Schrattenholzer, 2000). A stochastic version was developed to compare alternative approaches to risk modeling and to analyze the impact of hedging on transition pathways (Krey and Riahi, 2013; Messner et al., 1996).

[1] See the documentation of NEMS at www.eia.gov/outlooks/aeo/nems/documentation.

MESSAGE was frequently applied to pertinent questions at the interface between science and policy in the context of energy system transitions and environmental questions. In the past decade alone, the MESSAGE model was used in a number of highly visible projects: it underpinned a substantial part of the *Global Energy Assessment* (GEA, Riahi et al., 2012); it contributed one of the marker scenarios for the *Representative Concentration Pathways* (RCP, Riahi et al., 2011); and it was used as a marker scenario for the *Shared Socio-economic Pathways* (SSP, Fricko et al., 2017). Scenarios developed using the MESSAGE model were also included in the analysis of the *Intergovernmental Panel on Climate Change* (IPCC, Edenhofer et al., 2014, and earlier Assessment Reports).

Over the past years, the global version of the MESSAGE model has been extended to include many drivers relevant for the analysis of energy supply and demand: Rao and Riahi (2006) analyzed the role of multiple non-CO_2 greenhouse gases and implemented a detailed accounting of pollutants. Sullivan et al. (2013) and Johnson et al. (2016) developed methodologies to specifically represent the challenges presented by variable renewable electricity sources on the power system. Fricko et al. (2017) developed an endogenous integration with the land-use model *GLOBIOM* (Havlík et al., 2014) and the forestry model *G4M* (Kindermann et al., 2008).

Cameron et al. (2016) and Ekholm et al. (2010) extended MESSAGE to specifically incorporate the analysis of universal access to clean energy, which is a particular public health concern among the poor population in South East Asia. Lehtveer et al. (2015) and McCollum et al. (2013) embedded MESSAGE in a multi-criteria assessment framework to analyze the trade-off and synergies between different objectives including energy security, climate change mitigation and reduction of air pollution.

All of the studies listed above are based on the global MESSAGE implementation developed at IIASA, where the world is represented by eleven regions. The documentation of this MESSAGE version, including the integration with the GLOBIOM and MACRO models, is available at **data.ene.iiasa.ac.at/message-globiom/** (Krey et al., 2016); this documentation page will be updated regularly to incorporate new model developments and scientific publications.

1.2.1. MESSAGE at the International Atomic Energy Agency

Beyond the applications of MESSAGE at IIASA working on the science-policy interface, the *matrix generator* underpinning the MESSAGE model until now was also distributed by the International Atomic Energy Agency (IAEA) to its member countries for strategic energy planning purposes since 2000 (IAEA, 2016). It continues to be used actively in ca. 30 countries, and numerous model applications and publications at the national level have resulted from that collaboration, for example in Brazil (de Lucena et al., 2010; Herreras Martínez et al., 2015), Lithuania (Streimikiene and Balezentis, 2013) and Chile (Watts and Martinez, 2012).

1.3. An evaluation of the MESSAGE model

Wilson et al. (2017a) define five criteria according to which numerical systems assessment tools should be evaluated for their adequacy to yield policy-relevant insights: *appropriateness* for the research question; *interpretability* of the underlying concept and methodology as well as the results; *verifiability* and detailed documentation of the model code including both the mathematics and the scientific workflow (data management and post-processing); *credibility* of the derived insights and policy recommendations; and *usefulness* by advancing understanding of challenges and policy options.

Similar criteria for evaluation are, for example, discussed by Schwanitz (2013), stressing in particular that model evaluation must be treated as a continuous process, rather than a one-off exercise. Furthermore, comprehensive documentation and transparency are identified as paramount for any meaningful evaluation.

Jakeman et al. (2006) make a similar case, structured around 10 iterative steps to consider during model development, parametrization and evaluation/testing. The authors emphasize the need to start with a clear conceptualization of the system to be modeled and continue with a critical evaluation of the model structure and data collection techniques. Then, gaining an understanding of parameter uncertainty are as important as identifying criteria for verification and testing. Most importantly, assumptions and modeling choices made early in the development process need to be critically reviewed regularly.

Principles for the development and assessment of energy-system optimization models in particular (i.e., frameworks like MESSAGE) are presented by DeCarolis et al. (2017). They discuss in detail the many dimensions in which such models can be extended, including endogenous learning, price-elastic demand, and uncertainty. The authors highlight potential trade-offs between more advanced approaches and computational complexity, and explicitly caution against including too much detail when this is not pertinent for the research question at hand.

Given the extensive track record of the MESSAGE model, the framework has built up substantial credibility and has proven to be useful in the evaluation of systems transition pathways. Most of the extensions for energy-systems models discussed by DeCarolis et al. (2017) have been implemented in MESSAGE at some point, as laid out in the previous section. Where these extensions involved substantial additional computational burden, they have been used only in stand-alone branches; other developments were integrated into the main version. The versatility of MESSAGE is furthermore illustrated by the many developments beyond the energy sector added over the years, indicating that the combination of a least-cost systems optimization approach with a general-economy model is an appropriate avenue for many pertinent research questions at the science-policy interface.

The aspects in which the existing MESSAGE framework was lacking are related to transparency, interpretability and verifiability. The model code (a compiled *matrix generator* written in C) and the post-processor (Strubegger, 1984), while being continuously updated and improved over the years, do not meet software development standards of today. A flexible formulation of "user-defined relations" (linear constraints to the optimization problem) allowed to implement a broad range of mathematical features like share-constraints or bounds on aggregate activities across technologies – but the implementation did not always provide the required versatility or ensure straightforward interpretability of the model structure. Over the years, user-defined relations were used for so many purposes in the global MESSAGE model that documentation of constraints and parameter values became a burden.

Alas, the most significant drawback was a multi-layered, custom-built text-file format for input data, model output and processed results. This made it relatively cumbersome to keep track of data changes and hampered efficient data pre- and post-processing routines or integration with scientific programming workflows.

1.4. Goals for developing a new systems modeling framework

When setting out to work on a new platform based on the existing MESSAGE model, our aim was building on the proven track record – while developing a framework that specifically replaced those parts of the system limiting effective modeling. In particular, the goal was to allow more efficient scientific workflows and direct integration with external data sources and other models or tools. A second aim was providing an implementation where users could easily add new equations and parameters for specific use cases like representation of renewables or emissions accounting, rather than resorting to the generic "one-size-fits-all" implementation of constraints using the user-defined relations in the previous MESSAGE framework.

Framing our ambition in terms of the three paradigm shifts stated earlier, both the underlying mathematical formulation and the data processing environment should be optimized towards a comprehensive representation of *complex systems*. In particular, this includes native

handling of spatially explicit data sets or simple development of models that operate across different spatial and temporal scales.

Regarding the *integration of models* across sectors, methodologies and disciplines, the underlying principles of the platform anticipate that such linkages will be of paramount importance for any relevant analysis over the next years. The framework provides well-defined *interfaces* to facilitate the exchange of input data and model results across disciplines and methodologies.

To this end, all components apart from the actual equations of the MESSAGE$_{ix}$-MACRO model written in GAMS (see below) are implemented in a way that the platform can be used as a data warehouse and processing facility for any numerical model. The database architecture, the browser-based user interface, and the interfaces to the scientific programming languages Python and R are agnostic regarding the type of mathematical program or data.

Last, but not least, the platform should support best-practice collaborative research and facilitate transparency, interpretability, and verifiability. There are, as a minimum, three dimensions to the quest for openness in this context (cf. Pfenninger et al., 2017): The mathematical formulation and the implementation in a numerical programming language must be easily accessible. Furthermore, it should be simple to extend the structure or add model features (e.g., introduce new types of equations) even for non-expert users. Next, the underlying data and model results must be accessible in an intelligible format. Finally, the scientific workflows used for data pre-processing, model execution, and analysis of results must be implemented in a way that allows scrutiny and external evaluation.

As a consequence of these considerations, the platform was structured around four key features illustrated in Fig. 1. The ultimate goal, as elaborated previously, was to develop a framework to improve transparency and effectiveness of policy-relevant modeling. The individual building blocks are listed below and described in detail in the following sections.

- The MESSAGE$_{ix}$-MACRO mathematical model – A least-cost systems optimization program integrated with a stylized general-economy model implemented in GAMS and published under an open-source license (Section 4);
- A Java back-end linked to a powerful database architecture for scenario data management, incorporating comprehensive version control (Section 2.3);

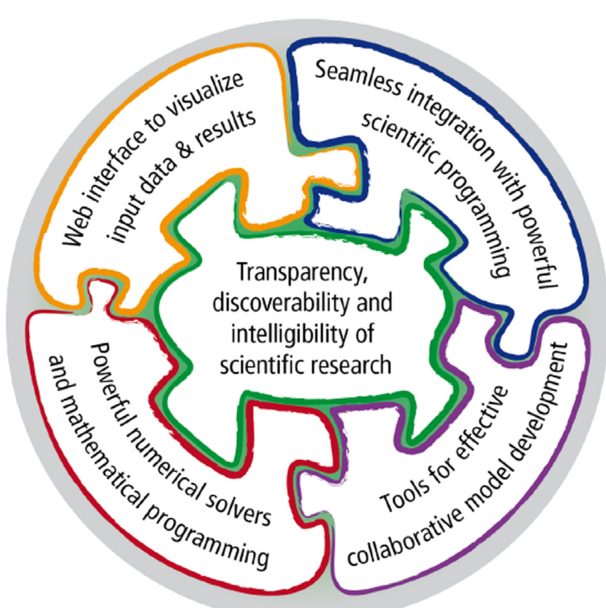

Fig. 1. Key features of the *ix modeling platform*.

- Interfaces to the scientific programming languages Python and R for data processing and implementation of workflows for model integration (Section 2.5);
- A REST API for standardized data exchange using web services (Section 2.6);
- A user interface accessible via any web browser for model/scenario management and data/results analysis, offering state-of-the-art features like Pivot table visualization and easy access to full data version control features including an item-by-item change log (Section 3).

In line with the goal of transparency and accessibility, the GAMS equations and the Java, Python and R interfaces are implemented with auto-documentation functionality: html pages are generated automatically with detailed documentation of the mathematical formulation for MESSAGE$_{ix}$ and MACRO, as well as a comprehensive manual of all functions implemented in the programming interfaces. All these documentation pages are generated directly from the model code, in line with best practice of software development, and they can be viewed with any current web browser.

We envision that the *ix modeling platform* and the MESSAGE$_{ix}$ model will be frequently upgraded to incorporate new features and support the integration with other models. Therefore, the following sections do not provide a comprehensive documentation of all features. Instead, this manuscript provides an overview of the guiding principles and key components, and we refer to the comprehensive documentation available at **MESSAGEix.iiasa.ac.at** and the source code of the respective packages published on GitHub for further (and up-to-date) information.

2. A platform for *integrated and cross-cutting* modeling

The *ix modeling platform* (ixmp) is a powerful and versatile data warehouse for reference data timeseries, modeling input, output from a numerical solver, and processed results. The framework is geared towards facilitating *integrated* and *cross-cutting* analysis – hence the name *ix modeling platform* for the entire framework and MESSAGE$_{ix}$ for the least-cost systems optimization model implemented on that platform. A detailed feature overview of the platform components and their interlinkage is shown in Fig. 2.

The platform is tailored to support scenario assessment in the sense often used in economic or systems analysis: develop a baseline scenario using business-as-usual assumptions or projections, and then compute the outcome under a large number of parameter variations. In the context of integrated assessment models, these scenario assumptions can include emission constraints, policy measures like taxes or subsidies, as well as different availability and characteristics of specific technologies or processes.

It is important to point out that no aspect of the platform is specific to integrated assessment models, linear optimization, or the MESSAGE framework (see Section 4). The entire data warehouse infrastructure was developed for generic model definitions and solution approaches. The platform can be used for linear optimization problems, computable general equilibrium models (e.g., the AIM/CGE model, Fujimori et al., 2014), game-theoretic partial-equilibrium approaches (e.g., Huppmann and Egging, 2014), as well as simulation and agent-based models (e.g., LEAP, Heaps, 2016 or COPA, Schmidt et al., 2016) – as long as they are derived from parameters and sets/mappings to define the model structure, and generate output that can be framed as variables and/or (marginals to) equations.

2.1. Package structure and license

The ixmp package consists of a compiled version of the Java core to connect to and work with a database instance, as well as the open-source interfaces to the scientific programming languages Python and R explained in more detail in the following sections. The package is

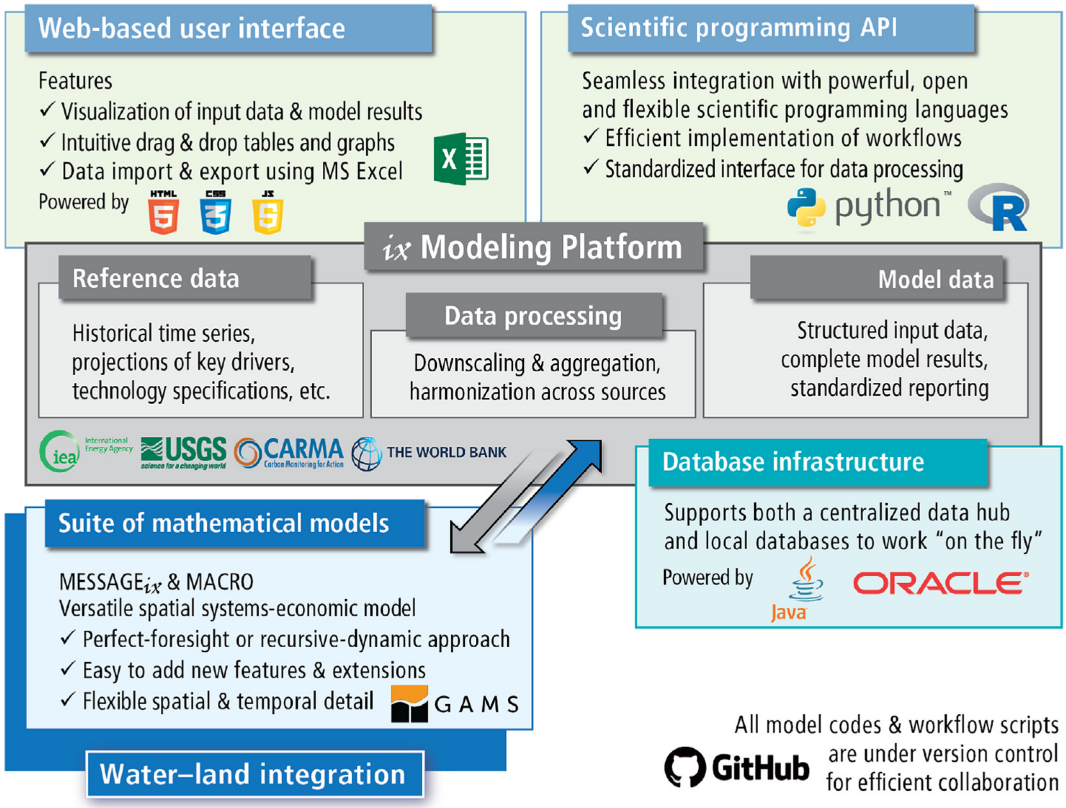

Fig. 2. Components and interlinkages of the *ix modeling platform*.

distributed under an *Apache License, Version 2*.[2]

The repository includes detailed installation instructions, a list of dependencies, as well as several tutorials and toy problems. Dantzig's transport problem is used to illustrate the model development and scenario analysis workflow, including the integration of the *ix modeling platform* with GAMS (see Section 2.7).

The package can be used as a library for economic/environmental modeling or scenario assessment as elaborated above. In line with the notion of "separation of concerns", the package covers the functionality and interfaces for data management; any mathematical equations, the original input data, and the actual data pre- and post-processing scripts should always be kept separate from the ixmp repository.

Contributions to the code base by users are encouraged. A *Contributor's License Agreement* (CLA) must be signed by all collaborators to establish a continuing open development of features and additional tools, while removing potential conflicts of attribution or licensing in the future.

Further information and download: **github.com/iiasa/ixmp**.

2.2. Tools & features for effective model development and analysis

Software development and data management have become an integral part of scientific work. While much computational work has been done on an ad-hoc basis in the past, there is increasing awareness for the need of applying professional standards to writing code and processing data (Heaton and Carver, 2015).

Given the constant pressure to publish results while they are relevant in a specific policy discourse or in time for project deadlines, comprehensive documentation and archiving of all pertinent data and tools are too often treated as an afterthought in scientific work. This puts reproducibility of the research and verifiability of the derived insights at risk. Useful overviews of "good-enough practices in scientific computing" are given by Wilson et al. (2017b) and Sandve et al. (2013); recommendations include breaking programming code into re-usable functions, using version control for any and all code, and applying comprehensive data management across the entire workflow.

The ixmp package implements these practices on two levels: first, the platform facilitates effective modeling and analysis in collaborative scientific work by providing a simple database solution for working with scenarios. This allows modelers to focus on the actual research question by automatically enforcing data version control and consistency checks of model data. The standardized interfaces also makes it easier to implement script-based workflows for scenario development and analysis, reducing the need for "manual" (and thereby error-prone) data processing.

Secondly, the implementation of the *ix modeling platform* itself follows best-practice of collaborative software engineering: the software development workflow uses GitHub for version control and all new code is reviewed by co-developers; comprehensive documentation pages are built directly from the source code; and the software is based on open-source packages. A number of these key features are described in more detail below; see the online documentation and the release notes for further details.

2.2.1. Management of scenario and reference data

The standard "block" of data in the platform is either a TimeSeries or a Scenario object, identified by a 'model' and 'scenario' name and tagged with a version number. Each object may contain either data from a reference source, which can be used as "raw" input data during model development or calibration, or processed model input data and numerical results for one scenario, projection, or pathway.[3]

[2] See www.apache.org/licenses/LICENSE-2.0 for more information.

[3] These terms are sometimes used interchangeably: for the purpose of this manuscript, we mean one numerical evaluation of a baseline or counterfactual

To illustrate the type of objects using examples from the realm of applied energy systems modeling, an object could contain the following data:

i) reference energy system data like consumption and production quantities from a source such as the International Energy Agency (IEA),[4] fossil reserves estimates collected by the US Geological Survey (USGS),[5] or timeseries data from a provider such as S&P Global Platts[6];
ii) processed results from any model following the IAMC convention explained below, like those submitted by modeling teams to the scenario database of the 5th Assessment Report of the IPCC (AR5)[7]; or
iii) all pre-processed input data (sets and parameters) as well as outputs and post-processed results for a numerical model instance.

The first two use cases are implemented by the class TimeSeries, while the third is implemented as a Scenario (see Table 1). The class Scenario extends (i.e., is a sub-class) of a TimeSeries.

2.2.2. Standardized reference data and reporting of results

Defining a common standard for reference data and processed model results that has sufficient flexibility to be applicable across many use cases, yet at the same time provides enough structure to be useful, is a constant challenge. Too often, developers of new tools define yet another comprehensive standard, only to realize limitations later. And even if that is not the case, such reporting standards rarely become implemented across multiple tools, models, or research groups. This limits interoperability of software and standardization of features.

To avoid a similar fate, the ixmp package implements the reporting template developed by the *Integrated Assessment Modeling Consortium* (IAMC).[8] This template has proven to be useful in multiple model intercomparison projects including the Shared Socioeconomic Pathways (SSP), the Representative Concentration Pathways (RCP), the 5th Assessment Report of the IPCC (AR5), and multiple rounds of the Energy Modeling Forum (EMF).[9] More importantly, a large number of research groups have already implemented routines and tools to export their native model outputs to the IAMC template. The community is also spending substantial effort to provide standardized tools and workflows to work with this template[10]

According to the template convention, any timeseries data is structured as follows:

model	scenario	region	variable	unit	< year1 >	< year2 >	…
…	…	…	…	…	…	…	…

Here, the columns “model” and “scenario” are determined by the model and scenario name identifying an TimeSeries or Scenario; columns “region” and “unit” are generic, though new terms have to be initialized via the Platform class (see Table 1).

The column “variable” provides the required versatility of this template: it can contain any string, and a semi-hierarchical structure can be implemented using the “|” character, e.g., Category|Subcategory|Subsubcategory|Additional Specification or to use a more tangible example from energy systems modeling:

Secondary Energy|Electricity|Wind|Onshore.

The number and sequence of sub-categories is completely generic, and no hierarchical structure is imposed by the ixmp package. This means that whether the sum over all subcategories (Category|Subcategory) matches the values provided for the “parent” timeseries (Category), if reported, is not validated by the platform. Instead, it is up to the modeling team (or teams in model intercomparison projects) to define variable trees as required for the specific analysis.

2.2.3. Model data consistency evaluation

The ixmp Java core contains multiple consistency checks for efficiently working with model data. Helpful error messages are raised via the scientific programming interfaces wherever possible, to help the user identify potential issues.

The following examples illustrate the data and workflow consistency checks implemented in the platform. They are intended to reduce mistakes from typos or common oversights during model development:

i) A new parameter can only be initialized over an index set if that set has been previously initialized. *Initializing a new parameter "a(i)" requires an index set "i"*
ii) Adding a new element to a parameter only works if that item is defined as an element of the respective index set(s) *Adding a new element "item" to parameter "a(i)" requires that "item" is an element of index set "i"*
iii) If a solution (levels/marginals of variables and equations reported by a numerical solver) has been imported to a Scenario, it is not possible to check out and edit any input data for that Scenario. This ensures that the input data and the solution are always in sync.[11]

2.2.4. Comprehensive version control and item-by-item change log

Another major challenge of scenario assessment is the crucial task of keeping track of input data and assumptions, as well as complete outputs and numerical results, across an oftentimes large ensemble of model runs. This is particularly important to guarantee reproducibility and assessability of modeling results. To facilitate this task, the *ix modeling platform* implements comprehensive data version control.

Each model/scenario instance is tagged by a version number, such that no data needs to be overwritten or deleted in the underlying database instance when updating a data point. For each model/scenario identifier, one version can be assigned as the *default* version.

The platform comes with standard functions to clone (i.e., copy, duplicate) an existing Scenario instance for use as starting point of scenario analysis or development of new baselines. The platform also implements a check-out/make-changes/commit workflow, to ensure that only one user at any time can make changes to one model/scenario/version instance in the database. All data changes are logged in the database with a user/timestamp, a commit message, and an item-by-item log including the previous value, allowing the user to “undo” any data changes.

The scientific workflow envisaged here is the following:

i) Develop and calibrate a Scenario
ii) Clone the baseline Scenario under a new model/scenario name
iii) Check out the Scenario, modify the model structure (i.e., sets and

(*footnote continued*)
trajectory to be used in conjunction with a mathematical model.
[4] See www.iea.org/statistics for an overview of IEA statistics and database.
[5] See **energy.usgs.gov** for an overview of datasets provided by USGS on energy and natural resources.
[6] See www.platts.com for more information.
[7] The AR5 scenario database is available at **tntcat.iiasa.ac.at/AR5DB/**.
[8] Refer to **data.ene.iiasa.ac.at/database** for details on the reporting template.
[9] See **emf.stanford.edu** for a list of current and past projects.
[10] See the public community repository at **github.com/IAMconsortium** for more information.

[11] In case a Scenario needs to be edited and solved again, the user can either remove the solution to allow data changes (remove_solution()) or create a copy/clone of the Scenario without the solution (clone()).

Table 1
Main object classes of the *ix modeling platform*.

Class name	Role and functions
Platform	Entry point or gateway for connecting to a particular database instance and retrieving or editing data. This class also provides functions to manage scenarios stored in a database instance, (e.g., to access meta-data including the last-edit user/timestamp) and add common terms like region names or units for reporting.
TimeSeries	Reference data from one source or processed model result from a scenario following the format of the *Integrated Assessment Modeling Consortium* (IAMC) timeseries data. column names: model, scenario, region, variable, unit, [years]
Scenario	Structured model instance input data (sets/parameters) and model output (variables/equations). Model results can be converted to the IAMC timeseries format and accessed using the functions of the TimeSeries class. A Scenario can be initialized with a specific *scheme*, which extends features of the generic class or automatically initializes a specific list of sets, parameters and variables.[a]

[a] As an example, creating a new Scenario using the scheme MESSAGE initializes all sets and parameters that are required by the GAMS implementation of MESSAGE$_{ix}$-MACRO, see Section 4. The scheme also adapts the function to_gdx(), such that additional information including the MESSAGE$_{ix}$ version number are written to the GAMS gdx data file.

mappings) and edit parameter values

iv) Commit the changes to the Platform (and underlying database instance)

v) Solve the model (export Scenario to a numerical solver using e.g., GAMS, solve the model instance, import the solution to the Scenario)

vi) Use the scientific programming interface (see Section 2.5) or the browser-based user interface (see Section 3) to analyze the results.

2.2.5. Continuous integration and collaborative development

All code underpinning the ixmp package is maintained on GitHub to facilitate collaborative development and code review (Petre and Wilson, 2014). This also simplifies distribution of updates to users. The package implements a number of *unit tests*.[12] These tests are executed on *CircleCI*[13] for any pull request issued via GitHub, as well as on a user's computer during installation of the ixmp package. They ensure that updates do not cause conflicts with existing features and that the package is installed correctly.

2.3. The Java core

The core component of the *ix modeling platform* is a Java Virtual Machine to manage the connection to an underlying database instance and retrieve or edit data. Java is a compiled language, which ensures fast access times, and there are established Application Programming Interfaces (API) from Java to Python, R, GAMS, and other programming languages.

Another advantage of a compiled program at the heart of the modeling platform is the definition of well-defined handshakes between API's and the data warehouse. This allows to implement very rigid consistency checks for data and access levels, as described briefly above and illustrated comprehensively in the online documentation and tutorials.

2.4. The database architecture

At the current status of development, the platform supports connections to *ORACLE* database servers for high-powered, large-scale scenario analysis, as well as local, file-based *HyperSQL DataBase* (HSQLDB) instances for development, testing, and scenario analysis at smaller model sizes.[14] Interfaces from the Java core to other relational database formats or even a switch to non-relational structures are possible extensions. This may become useful as the size of stored data increases and access time develops into a bottleneck constraining efficient modeling.

When initializing a Platform instance and connecting to a database (either a scheme on an ORACLE server or a local HSQLDB file), the database migration tool *flyway*[15] manages the creation or update (if necessary) of SQL tables required for the ixmp package. Using an out-of-the-box tool ensures that users do not need to have any knowledge of SQL database management for getting started. Also, future developments of ixmp that require additional tables in the SQL schema, like better metadata categorization and tagging features, can be rolled out and used with existing ixmp database versions with minimal overhead.

2.5. Interfaces to scientific programming languages: Python & R

To facilitate efficient model development and scenario analysis, it is paramount to establish effective integration with powerful scientific programming tools. This allows to use script-based workflows and automate many aspects of data pre- and results post-processing. However, the modeling community is split in roughly equal shares between R and Python as the programming language of choice, while Matlab and other options are also commanding a sizeable community. Often, the split runs within research groups, hindering easy collaboration and joint work using the same database.

Pursuing the aim to develop a general modeling platform that is useful to many different communities and across disciplines, the ixmp framework supports integration with Python and R. While the notation convention obviously differs between theses languages, all functions and interfaces to work with the platform are implemented in as parallel a manner as possible. Furthermore, the basic tutorials for working with the ixmp framework are provided in both programming languages.

2.5.1. The Python package ixmp

The interface between the ixmp Java core and Python is an installable package ixmp with several Python classes used as wrappers for the respective Java classes. It uses the Python package *Jpype*[16] for the connection between Python and the Java Virtual Machine. All functions include comprehensive auto-documentation and an API documentation page is built using *Sphinx*.[17]

The data transfer of sets, parameters, and the solution (levels and marginals of variables and equations at the optimum of an optimization problem) between the ixmp Java core and Python directly uses the Python pandas package for dataframes.

The package also includes several command-line interface (CLI) gateways, so that functions for importing timeseries data to a Platform instance and similar operations can be executed from any programming

[12] Unit tests are a method during software development, in which individual functions of a package or module are executed and compared to a pre-defined outcome. This ensures that adding new features or functions does not cause any existing parts of the software to break, and it facilitates code review by illustrating the intended application of a new feature.

[13] See www.circleci.com for more information.

[14] Refer to **hsqldb.org** for an overview and list of features.

[15] See **flywaydb.org** for more information.

[16] See **pypi.python.org/pypi/JPype1** for more information.

[17] See www.sphinx-doc.org for more information.

environment.

Further information: **MESSAGEix.iiasa.ac.at/api/python.html**.

2.5.2. The R package rixmp

The interface between the ixmp Java core and R is implemented as an installable R package rixmp. R packages are the fundamental unit of shareable code in R, and offer the advantage of bundling together code, documentation and tests, while making it easier to share with others (Wickham, 2015).

The package uses the rJava library and implements several *Reference Classes* for object-oriented programming as wrappers for the Java classes. Auto-documentation is generated using roxygen2[18] and is available through the R built-in help system.

An alternative option to the rixmp package is a connection to the core via the Python ixmp implementation using the R reticulate library.[19] We are currently exploring both options to assess which approach is preferable in terms of native support to users of the modeling platform and developing the Python and R interfaces in parallel.

Further information: **MESSAGEix.iiasa.ac.at/api/R.html**.

2.6. Data exchange via the REST API

The interfaces to the scientific programming languages Python and R introduced above are intended for high-powered computing and integration in a scientific workflow, with direct connection to the database. For other applications or for remote access to a server running ixmp and the database instance, the package includes a *Representational state transfer* (REST) web service, also called a REST API.

This service provides a generic and well-defined way to access and exchange data with a server running an ixmp instance, which is then itself connected to a database. The REST API can be used for any type of application, with mobile apps for smartphones and web applications running in a browser being the most common cases.

2.7. Integration with GAMS

The *ix modeling platform* includes an interface to the *General Algebraic Modeling System* (GAMS)[20], a software system for mathematical programming and large-scale numerical computation. The interface allows to export scenario data to a mathematical program implemented in GAMS and import the numerical solution to the platform instance after executing the program. In addition, the *ix modeling platform* includes a feature to create documentation pages from the GAMS code.

2.7.1. Data exchange between GAMS and ixmp

The data exchange is implemented using the GAMS-Java-API and uses the GAMS-specific format gdx.

Calling the function to_gdx() on a Scenario exports all input data to a gdx-file, which can then be imported by GAMS when executing a program (usually solving a mathematical problem). Upon completion of the numerical computation, GAMS can export the solution to another gdx-file, which is then imported to the Scenario using the function read_solution_from_gdx(). A wrapper function solve() combines these two functions together with a call to the GAMS command line interface to execute a mathematical program.

To illustrate the workflow of model development (i.e., defining the model structure, setting the parameter values) and solving the numerical instance using the ixmp-GAMS interface, the package includes a tutorial based on Dantzig's transport problem.[21] This example is often used as a “Hello World” equivalent for mathematical-programming languages.

Further information: **github.com/iiasa/ixmp/tree/master/tutorial/transport**.

2.7.2. GAMS code documentation using Python/Sphinx

Ensuring up-to-date and comprehensive documentation of mathematical equations, code, and workflow scripts is a constant challenge in software development. It is therefore understood as best-practice to keep the source code and a comprehensive documentation side-by-side, within the same document if possible. Because GAMS does not offer a suitable native documentation feature, we implemented an extension to write mark-up text directly in any GAMS code and translate it to html pages for a full-fledged manual.[22]

A parser scans GAMS files included in the package for documentation paragraphs. These are written in the *restructured text* (rst) format, an easy-to-read markup syntax that includes full LaTeX support for proper formulation of mathematical equations. The mark-up documentation paragraphs in the GAMS code are then processed using the Python package *Sphinx* (introduced above) to generate html pages viewable with any browser. These pages can be viewed offline and do not require a web server.

This feature is available as a stand-alone package under a *BSD-3* open-source license. It can therefore be included in any GAMS project independently of the ixmp package.

Further information and download: **github.com/iiasa/gams_stub/**

2.8. Outlook: other mathematical programming languages

At the time of writing, only the mathematical programming software GAMS has an explicit API implementation in ixmp to solve optimization problems, as described in the previous section. This is due to the parallel development of ixmp and the MESSAGE$_{ix}$ model (see Section 4).

Integration with programming tools in Python (e.g., *pyomo*[23]) or R[24] could be easily implemented via the respective API's. Furthermore, it is possible to implement data exchange features with other frameworks like the *GNU Linear Programming Kit* (GLPK).[25] Such functions would have to be added to the Java core, with respective implementation in the Python and R API's.

3. A web application for the modeling platform

The programming interfaces described in the previous section are complemented by a web application, which is a graphical user interface (GUI) accessible through any web browser. This reduces the barrier of entry and makes the ixmp framework accessible for technically less versed users.

The web application is connected to the REST API described in Section 2.6. Developing the web application using state-of-the-art libraries and toolkits allows to offer an appealing user experience with a focus on efficient data processing and intuitive visualizations.

The web application is still in development when submitting this manuscript. In the future, it will allow analyzing and visualizing model

[18] See roxygen.org/for more information.

[19] See **github.com/rstudio/reticulate**/for more information.

[20] See www.gams.com for more information and download.

[21] See www.gams.com/24.8/docs/userguides/userguide/_u_g_tutorial.html for the GAMS tutorial based on the transport problem.

[22] Since version 24.8, GAMS includes the extension model2tex, which parses mathematical equations and translates them to LaTeX. However, we found that this feature does not provide sufficient flexibility to add explanations and references to other parts of the code, and therefore implemented an alternative solution.

[23] See www.pyomo.org for an introduction and download.

[24] See **CRAN.R-project.org/view = Optimization** for an overview of optimization tools in R.

[25] See www.gnu.org/software/glpk for more information.

input data, optimization results and processed timeseries across multiple scenarios. It will also provide access to the scenario history and the detailed change-log of an ixmp database instance.

A beta version of the ixmp web application is used to run the *IAMC 1.5°C Scenario Explorer hosted by IIASA* (Huppmann et al., 2018).[26] At this state, the application provides intuitive visualization of scenario results. Fig. 3 shows two screenshots of that instance.

4. The MESSAGE$_{ix}$-MACRO integrated assessment model

The first model implemented within the ixmp framework is the MESSAGE$_{ix}$ model, a versatile systems optimization model based on the MESSAGE equations discussed in the introduction (Section 1.2). This model can natively be coupled with a single-sector computable general equilibrium (CGE) model (MACRO).

The MESSAGE$_{ix}$-MACRO energy system optimization and integrated assessment model is implemented in GAMS,[27] a powerful software system for mathematical programming and large-scale numerical computation. GAMS offers a good compromise between an easy-to-learn syntax with little programmatic overhead, and native integration with powerful numerical solvers including CPLEX and GUROBI. This allows even non-expert users to "look under the hood" and understand the mechanics of any equation, while providing the flexibility and numerical scalability to solve large-scale applications and easily extend beyond linear programming methods in future extensions.

4.1. Package structure, license, and documentation

The core component of the message_ix repository are the equations of the MESSAGE$_{ix}$ and MACRO models. The implementation and associated tools are distributed under an *Apache License, Version 2*, similar to the ixmp package. The installation instructions and dependencies are virtually identical to those of the ixmp package (see Section 2). The distribution also includes several tutorials to illustrate the development of a stylized national energy system model and scenario assessment for policy evaluation.

Comprehensive documentation pages compiled from the GAMS source code and the ixmp package documentation are publicly available at MESSAGEix.iiasa.ac.at. This website is updated automatically based on the most recent version of the public message_ix GitHub repository, using the GAMS documentation tool discussed in Section 2.7.2.

Further information and download: **github.com/iiasa/message_ix**.

4.2. The MESSAGE$_{ix}$ systems optimization model

The first version of the MESSAGE systems optimization model was developed in the eighties aimed at energy systems planning. The mathematical formulation and supporting tools developed over the past decades have been re-implemented and improved for public release together with this manuscript.

4.2.1. Purpose of the model

MESSAGE$_{ix}$ is a dynamic linear least-cost optimization problem. The model seeks to satisfy given *demand* levels by commodity and node (region, country, etc.) at minimal total cost. The objective function aggregates costs and expenditures across all modules detailed below. These include investment and operational costs for technologies, costs for exhaustible resource extraction and power generation from renewable energy sources, as well as emissions taxes and other expenditures.

The model determines the optimal configuration of the energy system (and other sectors as included in the underlying data, e.g., water supply) under various technical-engineering, socio-economic, or biophysical constraints. In the default mode, the model assumes *perfect foresight* until the end of the optimization horizon. *Myopic behavior or limited foresight* can also be implemented in a rolling-horizon solution approach.

4.2.2. Structure and modules

The mathematical formulation of MESSAGE$_{ix}$ is centered on *technologies* that use and produce/generate *commodities*. These commodities can be modeled at different *levels* to depict a reference energy system from primary extraction to consumption of final or useful energy. Specific aspects common in energy-environmental systems models such as *exhaustible and renewable resources* or *emissions* are implemented in dedicated modules.

4.2.3. Technologies

Technologies are the key building blocks of a model instance, representing steps along the supply chain of a reference energy(-environment- …) system. Each technology can have technical-engineering parameters such as upper and lower bounds on capacity or activity. They can also include dynamic constraints, in which the capacity expansion or activity level in one period constrains the feasible options at a later point in the model horizon. This can be used to represent the inertia in a system or economic-engineering limitations to the diffusion of a technology beyond a certain rate.

The MESSAGE$_{ix}$ implementation supports a detailed vintage representation of installed capacity for each technology. This allows to represent changing characteristics of an installation over time, like decreasing efficiency or increased operation-and-maintenance costs towards the end of a plant's technical lifetime. Due to the specific representation of both fixed costs (per unit of installed/maintained capacity) and variable/operating costs (per unit of activity), the model endogenously determines the optimal point in time for retiring a plant. This allows to distinguish between the "technical lifetime" of an installation (an engineering parameter) and the "economic lifetime", which is a model outcome depending on the market environment (i.e., when future revenue is lower than fixed and operating costs, making it cost-optimal to retire a plant).

4.2.4. Exhaustible resources

Non-renewable resources are usually the first level in a reference energy system, before the 'primary energy' level. To reflect the cost characteristics of fossil fuels and the typical extraction path across multiple deposits, commodities are distinguished by different *grades* in the mathematical formulation.

Each grade has an upper limit on cumulative extraction over the entire model horizon. In addition, constraints on total extraction in a period can be specified either in absolute terms (as an upper bound) or as a maximum share of remaining resources (i.e., initial endowment at the beginning of the model horizon less extraction in previous periods). These equations allow to implement the characteristic resource supply curves, where cheap deposits are used first, but not necessarily exclusively. If parametrized appropriately, the formulation ensures extraction from a range of deposits (or grades) in every period. The extraction then shifts gradually to costlier options over time or if a basin is depleted (see Fig. 4).

4.2.5. Renewable resources

Sullivan et al. (2013) proposed a mathematical formulation to appropriately incorporate the supply of renewable energy resources. Because high-quality locations for wind and solar power generation are usually exploited first, increasing penetration of these technologies results in increased capacity requirements per amount of energy generated. Renewables differ from non-exhaustible resources in this regard, where exploitation of deposits over time leads to increasing extraction costs.

[26] See **data.ene.iiasa.ac.at/iamc-1.5c-explorer/**for more information.

[27] See www.gams.com and Section 2.7.

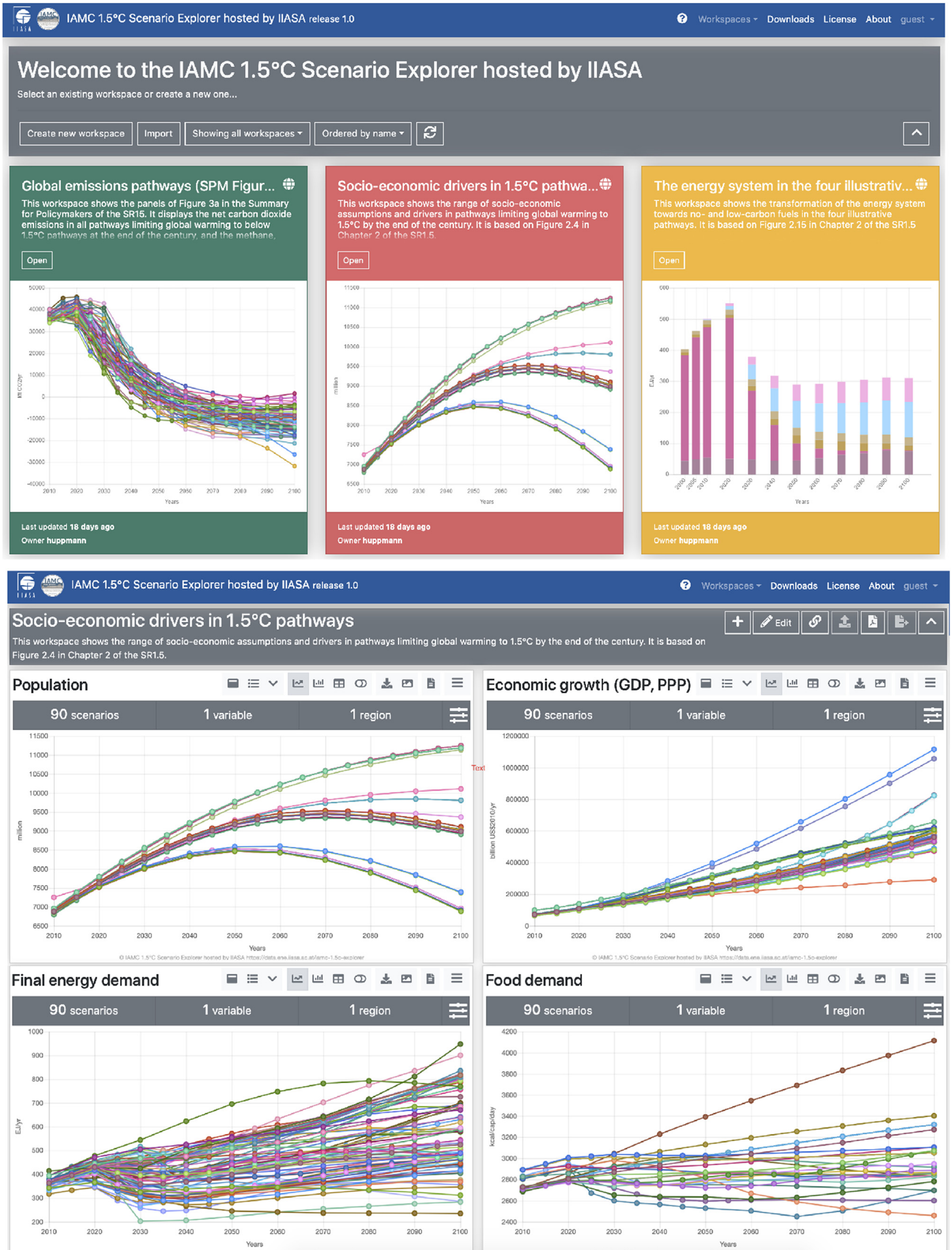

Fig. 3. Two screenshots from the *IAMC 1.5°C Scenario Explorer hosted by IIASA*(data.ene.iiasa.ac.at/iamc-1.5c-explorer, Huppmann et al., 2018), powered by a beta version of the ixmp web application (see Section 3).

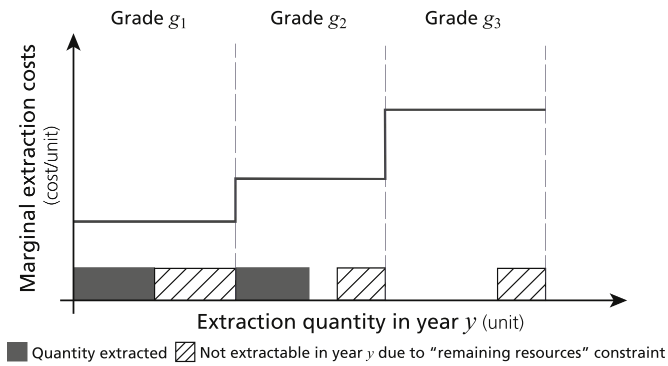

Fig. 4. Illustration of resource supply curves and the "remaining resources" constraint. The model cannot fully use the cheap resource grade g_1 in the illustrative period y, because only a fraction of each grade can be extracted per year. It has to partly switch to the more expensive resource grade g^2 to satisfy demand for this resource.

4.2.6. System integration, firm capacity, and flexibility

Energy-systems optimization models like MESSAGE$_{ix}$ represent the electricity sector in a rather stylized manner. This neglects the system-wide considerations from the portfolio of power plants and variability of renewable energy sources. Johnson et al. (2016) and Sullivan et al. (2013) proposed a number extensions to incorporate a better representation of the power sector in such reduced-form models.

The first consideration is the requirement for *flexibility*: each technology as well as load (demand) either provides energy that can be quickly ramped up or down (positive contribution to the flexibility metric), or increases the requirement to be offset by flexible plants (negative contribution). The aggregate of the flexibility metric over the electricity mix must be positive, ensuring that the system is "sufficiently flexible".

Another issue, often associated with variable renewable energy sources, are the impacts arising from high shares of variable and intermittent supply. Increasing penetration causes additional requirements on the system to accommodate variability and limited predictability. In such models, demand is usually represented as an average over a year or a sub-annual time slice. To guarantee that the *peak load* can be met at all times, the consumption of electricity multiplied by a *peak-load factor* must be offset by reliable and dispatchable power plants, often referred to as *firm capacity*. The formulation is illustrated in Fig. 5.

Due to their intermittent generation profile, installed capacity of variable renewable energy sources like wind and solar only counts towards firm capacity at a fraction, if at all. These considerations for renewables are modeled via multiple *ratings* of renewable resource categories, where the "quality" of power generation from these resources in terms of firm capacity and flexibility depends on their share in the total energy supply of a commodity.

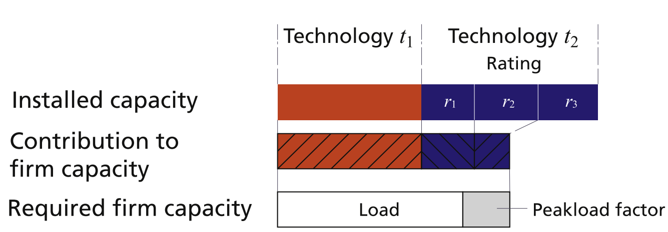

Fig. 5. Illustration of system integration constraints with regard to firm capacity: the capacity required to be "firm" (Johnson et al., 2016; Sullivan et al., 2013) depends on yearly average load and a peak-load mark-up; technology $t1$ is dispatchable and all installed capacity counts towards firm capacity; technology $t2$ depends on variable renewable energy source and different "ratings" counts towards firm capacity at a decreasing rate; the size of each rating bin depends on the share of that technology in the supply portfolio.

4.2.7. Emissions and pollutants

One common application of systems optimization models is the evaluation of pathways under greenhouse gas emissions constraints. The implementation therefore includes a dedicated formulation for upper bounds on emissions and pollutants. The implementation natively aggregates emissions across spatial scales (see also the following paragraph) such that an upper bound defined at the regional level constrains the total emissions from all subregions. The formulation is also flexible as to only account for emissions from specific categories of technologies or land-use scenarios, or to constraint the (average) emissions over a set of model periods.

4.2.8. Land-use model emulator

The land-use sector is of great importance in integrated assessment modeling both for providing bioenergy and food, and for acting as a source and sink of emissions. However, agriculture and forestry do not fit naturally into the formulation of energy system models based on technologies. Furthermore, merging two very detailed data sets of the energy system and land could create issues in terms of numerical computation and tractability.

For this reason, the MESSAGE$_{ix}$ implementation incorporates a generic land-use emulator, where the model can determine a linear combination of land-use scenarios or trajectories depending on cost characteristics, emissions profiles, output of commodities (e.g., crops, bio-energy), and input requirements (e.g., fertilizer).

The parametrization of the land-use scenarios is intended to be provided by other models specific to the agriculture sector. Hence, these equations can be characterized as a "model emulator", allowing to include a stylized representation of a full-fledged land-use representation in a MESSAGE$_{ix}$ model instance.

4.2.9. User-defined relations

When developing new features in a numerical model instance, it is often useful to provide a flexible way for introducing constraints or considering costs beyond those introduced in the respective sections in the objective function (technologies, resources, emissions, etc.). For this purpose, the MESSAGE$_{ix}$ formulation includes a generic implementation for (linear) constraints of the form $A\,x \leq b$. Here, the vector x represents the decision variables for capacity expansion, total installed/available capacity (summed over all vintages), and the activity level of a technology, and the matrix A are the multipliers for aggregating the "left-hand side" of the relation. In addition, the objective function includes a generic cost term of the form $(A\,x)^T c$, where $A\,x$ is the aggregate left-hand side of the relations as defined above, and c is a vector representing the costs.

Alas, user-defined relations should be replaced by specific equations and parameters when a new feature is "made permanent" and integrated into the core version of MESSAGE$_{ix}$. This will facilitate a continuously evolving documentation of model features and therefore support easy interpretation and parametrization for new collections of parameters and equations.

4.2.10. Native spatial and temporal disaggregation

As stated in the introduction and motivation for the development of the MESSAGE$_{ix}$ framework, an efficient treatment of spatial and temporal disaggregation levels is paramount to conducting relevant scenario analysis for future research questions. For this reason, the MESSAGE$_{ix}$ implementation includes a native consideration of technologies and other aspects across hierarchical spatial and time disaggregation levels.

In this way, it is possible to develop data sets for MESSAGE$_{ix}$ model instances where some technologies or commodities are considered at a regional and annual-average level, while other aspects are considered at a much finer spatial and/or temporal resolution level. One example for the first category of highly aggregated technology is coal extraction, where seasonal variability and local transport infrastructure are usually

of lesser concern. In contrast, power generation or water consumption along a river basin often require a high level of detail to accurately identify system-wide impacts and interdependencies across sectors.

This feature facilitates an effective scenario development process and reduces the processing burden on the numerical solvers, because only the relevant sectors are modeled at a high level of detail.

4.2.11. Infeasibility identification and debugging features

When developing large, complex systems models, it is often non-trivial and time-consuming to identify the reasons for infeasibilities or "model artefacts" (i.e., results caused by overly constraining or counter-acting model specifications). The MESSAGE$_{ix}$ implementation includes several checks during the GAMS pre-processing to identify potential data inconsistencies.

Furthermore, it includes the option to relax (all or a subset of) constraints and penalize the relaxation in the objective function. This allows to identify which constraints may cause infeasibilities of a scenario.

Further information: **MESSAGEix.iiasa.ac.at/debugging.html**.

4.3. The MACRO general-economy model

The aggregated, single-sector macro-economic model (MACRO) has been derived from the so-called *Global 2100* or *ETA-MACRO* model (Manne and Richeis, 1992), a predecessor of the MERGE model (Manne et al., 1995).

4.3.1. Purpose of the model

The MACRO model seeks to maximize the inter-temporal utility function of a single representative producer-consumer in each region. The decision variables are a sequence of savings, investment, and consumption levels. Capital stock, available labor and energy inputs jointly determine the total output of the economy according to a nested constant-elasticity-of-substitution (CES) production function.

4.4. Native integration of MESSAGE$_{ix}$ and MACRO

It is well-established that given the numerical scale of systems optimization problems and general-economy models, an iterative approach is most appropriate for integration of such types (Kypreos and Lehtila, 2015; Böhringer and Rutherford, 2009). The MESSAGE$_{ix}$-MACRO framework provides a native iterative procedure to incorporate the feedback from price changes on demand.

The integration includes several useful tools, including the calibration of parameters required for the MACRO model to the demand projections used for the corresponding MESSAGE$_{ix}$ data set. Because the iteration between the linear MESSAGE$_{ix}$ equations and the non-linear MACRO model may experience numerical problems, the iterative procedure also contains checks on oscillation of the solution. These features are described in detail as part of the framework documentation.

5. Summary and outlook

Numerical energy sector optimization and integrated assessment models are widely used in policy analysis and evaluation of pathways for transformation of the human and earth system. With the interaction of sustainable development and climate change gaining increasing prominence at the interface of science and policy, developing computational tools and models that operate across academic disciplines and methodologies becomes ever more important.

At the same time, reproducibility and transparency of scientific analysis are rapidly becoming a focus of researchers, policy-makers and funding agencies alike. New tools for effective collaboration from software development and business management are being applied in academia to facilitate interdisciplinary work and meet the growing demand for openness.

It is in this spirit that we re-implemented the well-established MESSAGE model, developed at IIASA over the past four decades. This manuscript presents the key building blocks of this new framework for *integrated and cross-cutting* analysis:

The framework is structured around a powerful database infrastructure to support effective modeling workflows and scenario management. This *ix modeling platform* (ixmp) is implemented as a versatile and flexible data warehouse, such that it can be used for any numerical simulation or optimization model. The package includes interfaces to the scientific programming languages Python & R for data processing and analysis, and to GAMS for large-scale numerical computation. A graphical user interface viewable with any web-browser provides state-of-the-art tools for data visualization and analysis of numerical results.

The framework includes an open-source GAMS implementation of the MESSAGE$_{ix}$ systems optimization model, where the subscript *ix* indicates that it is fully integrated with the ixmp package. The model allows to determine the optimal capacity portfolio and least-cost solution for satisfying a given demand for commodities or energy (services). The linear program can be natively coupled with the MACRO general-economy model to incorporate the feedback from prices on demand levels.

The MESSAGE$_{ix}$ model and the ixmp package are developed following commonly-agreed guidelines of best-practice for collaborative research and scientific software engineering. Comprehensive documentation pages of the mathematical formulation and the interfaces to Python & R are generated from the respective source codes. All scripts and tools are developed under version control on GitHub, and the framework is structured across multiple repositories to separate the mathematical formulation from the database infrastructure and scientific-programming interfaces. The repositories are distributed under the open-source Apache 2.0 license to encourage broad application and contribution from a wide range of users.

The versatile and modular structure of the ixmp package and the interfaces to widely used scientific programming languages allow to easily use this framework as a data warehouse for other numerical optimization, simulation or equilibrium models. Given the increasing need for integration of tools and methodologies across disciplines to better understand interdependencies and trade-offs, applying a common data warehouse architecture for multiple models will greatly facilitate such integration.

Going forward, it is our intention that the MESSAGE$_{ix}$ model and the ixmp package will be adopted by other research groups and applied modelers for energy sector planning and the analysis of transition pathways of human and earth systems in the context of climate change and sustainable development. The framework is structured specifically to facilitate a wide range of policy applications and extensions by adapting the mathematical formulation and developing new features of the ixmp package as required for specific research questions.

For that reason, this manuscript presented the structure of the framework and the inter-linkage of the building blocks, rather than a fully comprehensive documentation. Instead, we refer to the online documentation and user community pages for the latest releases, including the mathematical formulation, tutorials and ongoing developments.

Further information: **MESSAGEix.iiasa.ac.at**.

References

Böhringer, Christoph, Rutherford, Thomas F., 2009. Integrated assessment of energy policies: decomposing top-down and bottom-up. J. Econ. Dynam. Contr. 33 (9), 1648–1661.

Cameron, Colin, Pachauri, Shonali, Rao, Narasimha D., McCollum, David, Rogelj, Joeri, Riahi, Keywan, 2016. Policy trade-offs between climate mitigation and clean cook-stove access in South Asia. Nature Energy. https://doi.org/10.1038/nenergy.2015.10.

Cao, Karl-Kiên, Cebulla, Felix, Jonatan, J., Vilchez, Gómez, Mousavi, Babak, Prehofer, Sigrid, 2016. Raising awareness in model-based energy scenario studies—a transparency checklist. Energy, Sustainability and Society 6 (1), 28.

de Lucena, Andre Frossard Pereira, Schaeffer, Roberto, Salem Szklo, Alexandre, 2010.

Least-cost adaptation options for global climate change impacts on the Brazilian electric power system. Global Environ. Change 20 (2), 342–350.
DeCarolis, Joseph, Daly, Hannah, Dodds, Paul, Keppo, Ilkka, Li, Francis, McDowall, Will, Pye, Steve, Strachan, Neil, Trutnevyte, Evelina, Usher, Will, Winning, Matthew, Yeh, Sonia, Zeyringer, Marianne, 2017. Formalizing best practice for energy system optimization modelling. Appl. Energy 194, 184–198.
DeCarolis, Joseph F., Hunter, Kevin, Sreepathi, Sarat, 2012. The case for repeatable analysis with energy economy optimization models. Energy Econ. 34 (6), 1845–1853.
Edenhofer, Ottmar, Pichs-Madruga, Ramón, Sokona, Youba, Farahani, Ellie, Kadner, Susanne, Seyboth, Kristin, Adler, Anna, Baum, Ina, Brunner, Steffen, Eickemeier, Patrick, Kriemann, Benjamin, Schlömner, Steffen, Christoph von Stechow, Zwickel, Timm, Jan, C., Minx (Eds.), 2014. Climate Change 2014: Mitigation of Climate Change. Contribution of Working Group III to the Fifth Assessment Report of the Intergovernmental Panel on Climate Change. IPCC. Cambridge University Press, Cambridge, United Kingdom and New York, NY, USA..
Ekholm, Tommi, Krey, Volker, Pachauri, Shonali, Riahi, Keywan, 2010. Determinants of household energy consumption in India. Energy Pol. 38 (10), 5696–5707.
Fricko, Oliver, Havlik, Petr, Rogelj, Joeri, Klimont, Zbigniew, Gusti, Mykola, Johnson, Nils, Kolp, Peter, Strubegger, Manfred, Valin, Hugo, Amann, Markus, Ermolieva, Tatiana, Forsell, Nicklas, Herrero, Mario, Heyes, Chris, Kindermann, Georg, Krey, Volker, McCollum, David L., Obersteiner, Michael, Pachauri, Shonali, Rao, Shilpa, Schmid, Erwin, Schoepp, Wolfgang, Riahi, Keywan, 2017. The marker quantification of the Shared Socioeconomic Pathway 2: a middle-of-the-road scenario for the 21st century. Global Environ. Change 42, 251–267.
Fujimori, Shinichiro, Masui, Toshihiko, Matsuoka, Yuzuru, 2014. Development of a global computable general equilibrium model coupled with detailed energy end-use technology. Appl. Energy 128, 296–306.
Havlík, Petr, Valin, Hugo, Herrero, Mario, Obersteiner, Michael, Schmid, Erwin, Rufino, Mariana C., Mosnier, Aline, Thornton, Philip K., Böttcher, Hannes, Conant, Richard T., Frank, Stefan, Fritz, Steffen, Fuss, Sabine, Kraxner, Florian, Notenbaert, An, 2014. Climate change mitigation through livestock system transitions. Proc. Natl. Acad. Sci. Unit. States Am. 111 (10), 3709–3714.
Heaps, Charles G., 2016. Long-range Energy Alternatives Planning (LEAP) System. Stockholm Environment Institute, Somerville, MA, USA [Software version: 2017.0.11]. https://www.energycommunity.org.
Heaton, Dustin, Carver, Jeffrey C., 2015. Claims about the use of software engineering practices in science: a systematic literature review. Inf. Software Technol. 67 (Suppl. C), 207–219.
Herreras Martínez, Sara, Koberle, Alexandre, Rochedo, Pedro, Schaeffer, Roberto, Lucena, Andre, Szklo, Alexandre, Ashina, Shuichi, van Vuuren, Detlef P., 2015. Possible energy futures for Brazil and Latin America in conservative and stringent mitigation pathways up to 2050. Technol. Forecast. Soc. Change 98, 186–210.
Howells, Mark, Rogner, Holger, Strachan, Neil, Heaps, Charles, Huntington, Hillard G., Kypreos, Socrates, Hughes, Alison, Silveira, Semida, DeCarolis, Joe, Bazilian, Morgan, OSeMOSYS, Alexander Roehrl, 2011. The open source energy modeling system: an introduction to its ethos, structure and development. Energy Pol. 39 (10), 5850–5870.
Huntington, Hillard G., Weyant, John P., Sweeney, James L., 1982. Modeling for insights, not numbers: the experiences of the Energy Modeling Forum. Omega 10 (5), 449–462.
Huppmann, Daniel, Egging, Ruud, 2014. Market power, fuel substitution and infrastructure – a large-scale equilibrium model of global energy markets. Energy 75, 483–500.
Huppmann, Daniel, Kriegler, Elmar, Krey, Volker, Riahi, Keywan, Rogelj, Joeri, Rose, Steven K., Weyant, John, Bauer, Nico, Bertram, Christoph, Bosetti, Valentina, Calvin, Katherine, Doelman, Jonathan, Drouet, Laurent, Emmerling, Johannes, Frank, Stefan, Fujimori, Shinichiro, Gernaat, David, Grubler, Arnulf, Guivarch, Celine, Haigh, Martin, Holz, Christian, Iyer, Gokul, Kato, Etsushi, Keramidas, Kimon, Kitous, Alban, Leblanc, Florian, Liu, Jing-Yu, Löffler, Konstantin, Luderer, Gunnar, Marcucci, Adriana, McCollum, David, Mima, Silvana, Popp, Alexander, Sands, Ronald D., Sano, Fuminori, Strefler, Jessica, Tsutsui, Junichi, Van Vuuren, Detlef, Vrontisi, Zoi, Wise, Marshall, Zhang, Runsen, 2018. IAMC 1.5°C scenario explorer and data hosted by IIASA. In: Integrated Assessment Modeling Consortium & International Institute for Applied Systems Analysis.
IAEA, 2016. Modelling Nuclear Energy Systems with MESSAGE: a User's Guide. Nuclear Energy Series No. NG-T-5.2. International Atomic Energy Agency.
Jakeman, A.J., Letcher, R.A., Norton, J.P., 2006. Ten iterative steps in development and evaluation of environmental models. Environ. Model. Software 21 (5), 602–614.
Johnson, Nils, Strubegger, Manfred, McPherson, Madeleine, Parkinson, Simon C., Krey, Volker, Sullivan, Patrick, 2016. A reduced-form approach for representing the impacts of wind and solar PV deployment on the structure and operation of the electricity system. Energy Econ. 64, 651–664.
Keairns, D.L., Darton, R.C., Irabien, A., 2016. The energy-water-food nexus. Annual Review of Chemical and Biomolecular Engineering 7 (1), 239–262.
Kindermann, Georg, Obersteiner, Michael, Sohngen, Brent, Sathaye, Jayant, Andrasko, Kenneth, Rametsteiner, Ewald, Schlamadinger, Bernhard, Wunder, Sven, Beach, Robert, 2008. Global cost estimates of reducing carbon emissions through avoided deforestation. Proc. Natl. Acad. Sci. Unit. States Am. 105 (30), 10302–10307.
Krey, Volker, Riahi, Keywan, 2013. Risk hedging strategies under energy system and climate policy uncertainties. In: Kovacevic, Raimund M., Georg, Pflug, Teresa Vespucci, Maria (Eds.), Handbook of Risk Management in Energy Production and Trading. Springer US, Boston, MA, pp. 435–474.
Krey, Volker, Havlik, Petr, Fricko, Oliver, Johanna, Zilliacus, Gidden, Matthew, Strubegger, Manfred, Kartasasmita, Ina, Ermolieva, Tatiana, Forsell, Nicklas, Gusti, Mykola, Johnson, Nils, Kindermann, Georg, Kolp, Peter, McCollum, David, Pachauri, Shonali, Rao, Narasimha D., Rogelj, Joeri, Valin, Hugo, Obersteiner, Michael, Riahi, Keywan, 2016. MESSAGE–GLOBIOM 1.0 Documentation. International Institute for Applied Systems Analysis (IIASA), Laxenburg, Austria. http://data.ene.iiasa.ac.at/message-globiom/.
Kypreos, Socrates, Lehtila, Antti, 2015. Decomposing TIAM-MACRO to assess climatic change mitigation. Environ. Model. Assess. 20 (6), 571–581.
Lehtveer, Mariliis, Makowski, Marek, Hedenus, Fredrik, McCollum, David, Strubegger, Manfred, 2015. Multi-criteria analysis of nuclear power in the global energy system: assessing trade-offs between simultaneously attainable economic, environmental and social goals. Energy Strategy Reviews 8, 45–55.
Loulou, Richard, Labriet, Maryse, 2008. ETSAP-TIAM: the TIMES integrated assessment model. Comput. Manag. Sci. 5 (1–2), 7–40.
Manne, Alan, Mendelsohn, Robert, Richels, Richard, 1995. MERGE: a model for evaluating regional and global effects of GHG reduction policies. Energy Pol. 23 (1), 17–34.
Manne, Alan S., Richeis, Richard G., 1992. Buying Greenhouse Insurance: The Economic Costs of CO_2 Emission Limits. MIT Press.
Mantzos, Leonidas, Tobias, Wiesenthal, Anca Matei, Nicoleta, Rozsai, Mate, Navajas Cawood, Elena, Kourti, Ioanna, Papafragkou, Anastasios, Russ, Peter, Soria Ramirez, Antonio, 2016. POTEnCIA Model Description – Version 0.9. Joint Research Center, IPTS, Spain.
McCollum, David L., Krey, Volker, Riahi, Keywan, Kolp, Peter, Grubler, Arnulf, Makowski, Marek, Nakicenovic, Nebojsa, 2013. Climate policies can help resolve energy security and air pollution challenges. Climatic Change 119 (2), 479–494.
Meinshausen, Malte, Raper, S.C.B., Wigley, T.M.L., 2011. Emulating coupled atmosphere-ocean and carbon cycle models with a simpler model, MAGICC6 – Part 1: model description and calibration. Atmos. Chem. Phys. 11 (4), 1417–1456 (ACP).
Messner, Sabine, 1997. Endogenized technological learning in an energy systems model. J. Evol. Econ. 7 (3), 291–313.
Messner, Sabine, Schrattenholzer, Leo, 2000. MESSAGE–MACRO: linking an energy supply model with a macroeconomic module and solving it iteratively. Energy 25 (3), 267–282.
Messner, Sabine, Strubegger, Manfred, 1995. User's Guide for MESSAGE III. IIASA Working Paper WP-95-069.
Messner, Sabine, Golodnikov, Alexandr, Gritsevski, Andrii, 1996. A stochastic version of the dynamic linear programming model MESSAGE III. Energy 21 (9), 775–784.
Morrison, Robbie, 2018. Energy system modeling: public transparency, scientific reproducibility, and open development. Energy Strategy Reviews 20, 49–63.
Paltsev, Sergey, 2017. Energy scenarios: the value and limits of scenario analysis. Wiley Interdisciplinary Reviews: Energy Environ. 6 (4).
Petre, Marian, Wilson, Greg, 2014. Code review for and by scientists. ArXiv:1407.5648. https://arxiv.org/abs/1407.5648.
Pfenninger, Stefan, Hawkes, Adam, Keirstead, James, 2014. Energy systems modeling for twenty-first century energy challenges. Renew. Sustain. Energy Rev. 33, 74–86.
Pfenninger, Stefan, DeCarolis, Joseph, Hirth, Lion, Quoilin, Sylvain, Staffell, Iain, 2017. The importance of open data and software: is energy research lagging behind? Energy Pol. 101, 211–215.
Pfenninger, Stefan, Hirth, Lion, Ingmar Schlecht, Schmid, Eva, Wiese, Frauke, Brown, Tom, Davis, Chris, Gidden, Matthew, Heinrichs, Heidi, Heuberger, Clara, Hilpert, Simon, Krien, Uwe, Matke, Carsten, Nebel, Arjuna, Morrison, Robbie, Müller, Berit, Pleßmann, Guido, Reeg, Matthias, Richstein, Jörn C., Abhishek Shivakumar, Staffell, Iain, Tim Tröndle, Wingenbach, Clemens, 2018. Opening the black box of energy modelling: strategies and lessons learned. Energy Strategy Reviews 19, 63–71.
Pietzcker, Robert C., Ueckerdt, Falko, Carrara, Samuel, de Boer, Harmen Sytze, Després, Jacques, Fujimori, Shinichiro, Johnson, Nils, Kitous, Alban, Scholz, Yvonne, Sullivan, Patrick, Luderer, Gunnar, 2017. System integration of wind and solar power in integrated assessment models: a cross-model evaluation of new approaches. Energy Econ. 64, 583–599.
Pindyck, Robert S., 2013. Climate change policy: what do the models tell us? J. Econ. Lit. 51 (3), 860–872.
Rao, Narasimha D., Riahi, Keywan, Grubler, Arnulf, 2014. Climate impacts of poverty eradication. Nat. Clim. Change 4, 749–751.
Rao, Shilpa, Riahi, Keywan, 2006. The role of non-CO_2 greenhouse gases in climate change mitigation: long-term scenarios for the 21st century. Energy J. 3, 177–2000 Special Issue.
Riahi, Keywan, Rao, Shilpa, Krey, Volker, Cho, Cheolhung, Chirkov, Vadim, Fischer, Guenther, Kindermann, Georg, Nakicenovic, Nebojsa, Rafaj, Peter, 2011. RCP 8.5—a scenario of comparatively high greenhouse gas emissions. Climatic Change 109 (1), 33.
Riahi, Keywan, Frank, Dentener, Gielen, Dolf, Grubler, Arnulf, Jewell, Jessica, Klimont, Zbigniew, Krey, Volker, McCollum, David, Pachauri, Shonali, Rao, Shilpa, van Ruijven, Bas, van Vuuren, Detlef P., Wilson, Charlie, 2012. Chapter 17 - energy pathways for sustainable development. In: Global Energy Assessment - toward a Sustainable Future. Cambridge University Press, Cambridge, UK and New York, NY, USA, pp. 1203–1306 and the International Institute for Applied Systems Analysis, Laxenburg, Austria.
Riahi, Keywan, van Vuuren, Detlef P., Kriegler, Elmar, Edmonds, Jae, O'Neill, Brian C., Fujimori, Shinichiro, Bauer, Nico, Calvin, Katherine, Dellink, Rob, Fricko, Oliver, Lutz, Wolfgang, Popp, Alexander, Cuaresma, Jesus Crespo, Kc, Samir, Leimbach, Marian, Jiang, Leiwen, Kram, Tom, Rao, Shilpa, Emmerling, Johannes, Ebi, Kristie, Hasegawa, Tomoko, Havlik, Petr, Humpenöder, Florian, Da Silva, Lara Aleluia, Smith, Steve, Stehfest, Elke, Bosetti, Valentina, Eom, Jiyong, Gernaat, David, Masui, Toshihiko, Rogelj, Joeri, Strefler, Jessica, Drouet, Laurent, Krey, Volker, Luderer, Gunnar, Harmsen, Mathijs, Takahashi, Kiyoshi, Baumstark, Lavinia, Doelman, Jonathan C., Kainuma, Mikiko, Klimont, Zbigniew, Marangoni, Giacomo, Lotze-Campen, Hermann, Obersteiner, Michael, Tabeau, Andrzej, Tavoni, Massimo, 2017. The Shared Socioeconomic Pathways and their energy, land use, and greenhouse gas

emissions implications: an overview. Global Environ. Change 42, 153–168.
Rosen, Richard A., Guenther, Edeltraud, 2015. The economics of mitigating climate change: what can we know? Technol. Forecast. Soc. Change 91, 93–106.
Sandve, Geir Kjetil, Nekrutenko, Anton, Taylor, James, Hovig, Eivind, 2013. Ten simple rules for reproducible computational research. PLoS Comput. Biol. 9 (10), e1003285.
Schmidt, Johannes, Cancella, Rafael, Pereira, Amaro O., 2016. An optimal mix of solar PV, wind and hydro power for a low-carbon electricity supply in Brazil. Renew. Energy 85, 137–147.
Schrattenholzer, Leo, 1981. The Energy Supply Model MESSAGE. IIASA Research Report.
Schwanitz, Valeria Jana, 2013. Evaluating integrated assessment models of global climate change. Environ. Model. Software 50 (Suppl. C), 120–131.
Sohail, Ahmad, Pachauri, Shonali, Creutzig, Felix, 2017. Synergies and trade-offs between energy-efficient urbanization and health. Environ. Res. Lett. 12 (11), 114017.
Streimikiene, Dalia, Balezentis, Tomas, 2013. Multi-objective ranking of climate change mitigation policies and measures in Lithuania. Renew. Sustain. Energy Rev. 18 (Suppl. C), 144–153.
Strubegger, Manfred, 1984. User's Guide for the Post-Processor of Message II. IIASA Working Paper-84-072.
Sullivan, Patrick, Krey, Volker, Riahi, Keywan, 2013. Impacts of considering electric sector variability and reliability in the MESSAGE model. Energy Strategy Reviews 1 (3), 157–163.
Ueckerdt, Falko, Brecha, Robert, Luderer, Gunnar, Sullivan, Patrick, Schmid, Eva, Bauer, Nico, Böttger, Diana, Pietzcker, Robert, 2015. Representing power sector variability and the integration of variable renewables in long-term energy-economy models using residual load duration curves. Energy 90, 1799–1814.
Watts, David, Martinez, Victor, 2012. Long-run energy and emissions modeling in Chile: scenario assessment using MESSAGE. IEEE Latin America Transactions 10 (2), 1525–1536.
Weyant, John, 2014. Integrated assessment of climate change: state of the literature. J. Benefit-Cost Anal. 5 (3), 377–409.
Wickham, Hadley, 2015. R Packages, first ed. O'Reilly Media, Inc.1491910593 9781491910597.
Wilson, Charlie, Kriegler, Elmar, van Vuuren, Detlef P., Guivarch, Celine, Frame, Dave, Krey, Volker, Osborn, Timothy J., Jana Schwanitz, Valeria, Thompson, Erica L., 2017a. Evaluating process-based Integrated Assessment Models of climate change mitigation. IIASA Working Paper 17.
Wilson, Greg, Bryan, Jennifer, Cranston, Karen, Kitzes, Justin, Nederbragt, Lex, Teal, Tracy K., 2017b. Good enough practices in scientific computing. PLoS Comput. Biol. 13 (6).